

Patent Application for
"SYSTEMS AND METHODS FOR RESOURCE MANAGEMENT IN INFORMATION
STORAGE ENVIRONMENTS"

Inventors: Chaoxin C. Qiu, Umesh Gupta, Scott C. Johnson, Sarma Kolavasi, Theodore S.
Webb, Richard W. Yu, and Mark J. Conrad

This application claims priority from co-pending United States Patent Application Serial Number 09/879,810 filed on June 12, 2001 which is entitled "SYSTEMS AND METHODS FOR PROVIDING DIFFERENTIATED SERVICE IN INFORMATION MANAGEMENT ENVIRONMENTS," and also claims priority from co-pending Provisional Application Serial No. 60/285,211 filed on April 20, 2001 which is entitled "SYSTEMS AND METHODS FOR PROVIDING DIFFERENTIATED SERVICE IN A NETWORK ENVIRONMENT," and also claims priority from co-pending Provisional Application Serial No. 60/291,073 filed on May 15, 2001 which is entitled "SYSTEMS AND METHODS FOR PROVIDING DIFFERENTIATED SERVICE IN A NETWORK ENVIRONMENT," the disclosures of each of the forgoing applications being incorporated herein by reference. This application also claims priority from co-pending United States Patent Application Serial No. 09/797,198 filed on March 1, 2001 which is entitled "SYSTEMS AND METHODS FOR MANAGEMENT OF MEMORY," and also claims priority from co-pending United States Patent Application Serial No. 09/797,201 filed on March 1, 2001 which is entitled "SYSTEMS AND METHODS FOR MANAGEMENT OF MEMORY IN INFORMATION DELIVERY ENVIRONMENTS," and also claims priority from co-pending Provisional Application Serial No. 60/246,445 filed on November 7, 2000 which is entitled "SYSTEMS AND METHODS FOR PROVIDING EFFICIENT USE OF MEMORY FOR NETWORK SYSTEMS," and also claims priority from co-pending Provisional Application Serial No. 60/246,359 filed on November 7, 2000 which is entitled "CACHING ALGORITHM FOR MULTIMEDIA SERVERS," the disclosures of each of the forgoing applications being incorporated herein by reference. This application also claims priority from co-pending United States Patent Application Serial Number 09/797,200 filed on March 1, 2001 which is entitled "SYSTEMS AND METHODS FOR THE DETERMINISTIC MANAGEMENT OF INFORMATION" which itself claims priority from Provisional Application Serial No. 60/187,211 filed on March 3, 2000 which is entitled "SYSTEM AND

APPARATUS FOR INCREASING FILE SERVER BANDWIDTH," the disclosures of each of the forgoing applications being incorporated herein by reference. This application also claims priority from co-pending Provisional Application Serial No. 60/246,401 filed on November 7, 2000 which is entitled "SYSTEM AND METHOD FOR THE DETERMINISTIC DELIVERY OF DATA AND SERVICES," the disclosure of which is incorporated herein by reference.

BACKGROUND OF THE INVENTION

The present invention relates generally to information management, and more particularly, to resource management in information delivery environments.

In information system environments, files are typically stored by external large capacity storage devices, such as storage disks of a storage area network ("SAN"). To access or "fetch" data stored on a conventional storage disk typically requires a seek operation during which a read head is moved to the appropriate cylinder, a rotate operation during which the disk is rotated to position the read head at the beginning of desired sectors, and a transfer operation during which data is read from the disk and transferred to storage processor memory. Time is required to complete each of these operations, and the delay in accessing or fetching data from storage is equal to the sum of the respective times required to complete each of the seek, rotate and transfer operations. This total delay encountered to fetch data from a storage device for each input/output operation ("I/O"), *e.g.*, each read request, may be referred to as "response time." "Service time" refers to a logical value representing the total time interval during which a request for information or data is receiving service from a resource such as a processor, CPU or storage device.

With most modern disk storage devices, the time required for data transfer to memory is typically smaller than the delay encountered when completing seek and rotate operations. Due to the large number of files typically stored on modern disk storage devices, the time required to fetch a particular file from storage media to storage processor memory is often a relatively time consuming process compared to the time required to transmit or send the file from memory on to other network devices. In the case of delivery of continuous streaming content (*e.g.*, delivery of large streaming multimedia video/audio files), service time is often particularly significant due to

the large number of I/O (e.g., read request) operations associated with such continuous files. For the delivery of such streaming files, it is particularly desirable to optimize system throughput performance and to provide quality control for delivered content.

5 In the past, efforts have been made to improve storage system performance and overcome the capacity gap between the main memory and storage devices. For example, caching designs have been formulated in an attempt to re-use fetched data effectively so as to reduce the workload of the storage system. Batch scheduling efforts have focused on attempts to use less storage resources to serve a maximum number of client requests. However, caching and batch
10 scheduling techniques do not directly address storage device behavior. Also considered have been data placement techniques that attempt to reduce seek time by placing data according to its access pattern, including both single disk block placement techniques and multi-disk file placement techniques.

15 Other efforts at enhancing system performance have been directed towards performance disk arm I/O scheduling. Disk arm I/O scheduling relates to knowledge of physical data location and I/O request priority/dead-lines. Examples of conventional disk arm I/O scheduling techniques include round-based scheduling such as round-robin (*i.e.*, first-come-first-serve), "SCAN" (*i.e.*, moving disk arm from the edge to the center and back), Group Sweeping
20 Scheduling ("GSS") (*i.e.*, partitioned SCAN), and fixed transfer size scheduling such as SCAN Earliest Deadline First ("SCAN-EDF") (*i.e.*, deadline aware SCAN).

Prefetching techniques implemented at both host and drive levels have also been proposed in an attempt to enhance system performance. In this regard, storage level prefetching
25 relates to attempts to improve efficiency of storage resource use, and application level prefetching relates to attempts to smooth variable-bit-rate traffic in multimedia applications. Aggressive prefetching may also include buffer sharing techniques to maximize the performance improvement.

30 When continuous content serving requirements exceed capability of storage resources and/or buffer memory capacity, viewers may experience "hiccups" or disruptions in the